Article

Granular Elastic Network Regression with Stochastic Gradient Descent

Linjie He [1], Yumin Chen [1,*], Caiming Zhong [2] and Keshou Wu [1]

1   College of Computer Science and Technology, Xiamen University of Technology, Xiamen 361024, China; linjiehe_jason@foxmail.com (L.H.); kswu@xmut.edu.cn (K.W.)
2   College of Science and Technology, Ningbo University, Ningbo 315211, China; zhongcaiming@nbu.edu.cn
*   Correspondence: cym0620@163.com

Abstract: Linear regression is the use of linear functions to model the relationship between a dependent variable and one or more independent variables. Linear regression models have been widely used in various fields such as finance, industry, and medicine. To address the problem that the traditional linear regression model is difficult to handle uncertain data, we propose a granule-based elastic network regression model. First we construct granules and granular vectors by granulation methods. Then, we define multiple granular operation rules so that the model can effectively handle uncertain data. Further, the granular norm and the granular vector norm are defined to design the granular loss function and construct the granular elastic network regression model. After that, we conduct the derivative of the granular loss function and design the granular elastic network gradient descent optimization algorithm. Finally, we performed experiments on the UCI datasets to verify the validity of the granular elasticity network. We found that the granular elasticity network has the advantage of good fit compared with the traditional linear regression model.

1. Introduction

In the 1970s, Zadeh, an American automatic control expert, introduced the concept of fuzzy sets [1]. Fuzzy sets and fuzzy logic have attracted a large number of scholars worldwide to research on the topic. Hobbs proposed the concept of granularity [2], and which is consistent with human cognition, memory, reasoning, and logic. After this, Zadeh introduced the concept of fuzzy information granulation [3]. He stated that information granulation is the division of a whole into many information granules, after which those granules are studied. In 1999, T.Y. Lin proposed granular computing [4] on this base and applied it to the field of data mining [5]. Skowron proposed a granular calculation method based on rough set theory [6]. Liu [7] constructed an approach to information granule reasoning from the perspective of fuzzy logic. Then, in 2001, Yao defined the neighborhood system [8] and studied the connection between hierarchical granulation and rough set approximation. In 2002, Miao proposed a knowledge-based approach to granular computing [9]. Then Miao [10] reconstructed the granular computing structure from the perspective of set theory. Wang et al. applied granular computing to big data information processing [11]. Chen [12] et al. presented a PS-tree based rough set feature selection method. In 2014, Qian reduced the computation time of the attribute approximate reduction problem by using parallel granular computation [13]. After that the neighborhood system granular structure, distance, and metric were proposed by Chen [14]. Chen further introduced the rough granular KNN algorithm. Li [15] combined the boosting algorithm with the granular KNN algorithm based on Chen's work [14] to further improve the performance of the granular KNN algorithm. Although different scholars have different

understandings of the definition of granules, the structural idea of granules has attracted more and more scholars to conduct research. Thus granular computing has been applied to a variety of fields, including information systems [16], image processing [17], cognitive systems [18], attribute selection [19,20], multi-attribute decision making [21], and other fields [22–26].

Machine learning is a general data processing technique that is divided into unsupervised learning and supervised learning. Unsupervised learning includes PCA, local linear embedding, K-means clustering, and other algorithms. Supervised learning mainly includes K-nearest neighbor, decision tree, linear regression, logistic regression, support vector machine, etc. Regression models are supervised learning modalities including linear regression, ridge regression, lasso regression, and elastic network regression. Linear regression [27] simulates the relationship between a dependent variable and one or more independent variables using the linear function. By averaging the weights by adding L2 penalty, ridge regression [28] can improve the fitting ability of regression models in pathological data. The lasso [29] regression can suppress the overfitting of linear regression to some extent, and the lasso regression obtains a sparse model by adding L1 regularization to make some weight coefficients zero. In contrast, elastic networks combine ridge regression and lasso regression to both sparse the weights and shrink the weight coefficients [30]; however, these traditional regression models struggle to deal with uncertainty data and set data. Granular computing is an effective tool to deal with uncertain data; therefore, we construct a granular-based regression model of elastic networks. The contributions of this paper are as follows.

First, we propose a granulation approach based on feature distance and define various granulation operations and granularity measure approaches. Secondly, we construct the granular elastic network regression system and give the loss function of granular elastic network. Similarly, we optimize the granular elastic network regression equation by stochastic gradient descent. Finally, we conducted experiments on the UCI datasets to verify the effectiveness of the granular elastic network and found that the granular elastic network has a better fitting performance than the traditional regression models.

This article consists of the following parts. The first chapter briefly introduces the granular computations and regression algorithms. Then, we define the granularity approach in Section 2. In Section 3, we propose the granular elastic network regression and further propose the granular elastic network optimizer and its optimization algorithm. After that, we show the experimental results in Section 4. Finally, we conclude the whole paper in Section 5.

2. Granulation

Fuzzy sets are an effectual tool for dealing with uncertain information. For the information system $U = (X, C, D)$, $X = \{x_1, x_2, x_3, \ldots, x_n\}$ is the sample set, $C = \{c_1, c_2, c_3, \ldots, c_m\}$ is the feature set corresponding to the sample, and D is the decision set corresponding to the sample. For the given sample $x \in X$, where the single feature $c \in C$, $v(x, c) \in [0, 1]$ denotes the value of the sample x normalized on the feature c.

Definition 1. *In the data set $U = (X, C, D)$, samples x_1, $x_2 \in X$, and a single feature $c \in C$. The distance between x_1 and x_2 on the feature c is:*

$$s_c(x_1, x_2) = |v(x_1, c) - v(x_2, c)|. \tag{1}$$

Definition 2. *For the data set $U = (X, C, D)$, we granulate the sample $x \in X$ on the feature $c \in C$. The sample x is granulated on feature c and forms a granule, which is defined as feature c to form a fuzzy granule, which is defined as:*

$$g_c(x) = \{g_c(x)_j\}_{j=1}^k = \{r_j\}_{j=1}^k = \{r_1, r_2, \ldots, r_k\}, \tag{2}$$

$$where\ r_j = S_c(x, x_j) = |v(x, c) - v(x_j, c)|. \tag{3}$$

r_j is the distance between sample x and sample x_j on feature c. It is easy to know from Definition 1 that $r_j = s_c(x, x_j) \in [0, 1]$. We define $g_c(x)$ as the granule and $g_c(x)_j$ as the *jth* granule kernels of the granule $g_c(x)$, and the granule consists of the granule kernels.

Definition 3. *For the data set $U = (X, C, D)$, there is any $x \subseteq X$ and any feature subset $B = \{b_1, b_2, b_3, ..., b_m\} \subseteq X$. The granular vector of x on the feature subset B is:*

$$G_B(x) = (g_{b1}(x), g_{b2}(x), g_{b3}(x), \ldots, g_{bm}(x))^T. \tag{4}$$

$g_{bm}(x)$ is the granule of x on the feature b_m. The granular vector is made up of granules, and the granules are consisted of granular kernels. So the granular vector can also be represented by the granular kernel matrix as follows.

$$G(x) = \begin{bmatrix} g_1(x)_1 & g_1(x)_2 & \cdots & g_1(x)_k \\ g_2(x)_1 & g_2(x)_2 & \cdots & g_2(x)_k \\ \cdots & \cdots & \cdots & \cdots \\ g_m(x)_1 & g_m(x)_2 & \cdots & g_m(x)_k \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1k} \\ r_{21} & r_{22} & \cdots & r_{2k} \\ \cdots & \cdots & \cdots & \cdots \\ r_{m1} & r_{m2} & \cdots & r_{mk} \end{bmatrix} \tag{5}$$

The granular vector $g(x)$ is a set consisting of granules, while the elements of a conventional vector are real numbers.

Definition 4. *For the data set $U = (X, C, D)$, there is any $x \subseteq X$ and the decision set $d_x \subseteq D$. The extended decision granule of x is:*

$$Y(x) = \{d_x, d_x, ..., d_x\}, \tag{6}$$

where the number of the elements is the size of X.

Example 1. *The following is an example of granulation.*

As shown in Table 1, the sample set $X = \{x_1, x_2, x_3, x_4\}$ is granulated on feature a as:

$$g_a(x_1) = \{0, 0.65, 0.6, 0.3\}, g_a(x_2) = \{0.65, 0, 0.05, 0.35\}, g_a(x_3) = \{0.6, 0.05, 0, 0.3\}, g_a(x_4) = \{0.3, 0.35, 0.3, 0\}. \tag{7}$$

The sample set $X = \{x_1, x_2, x_3, x_4\}$ is granulated on feature b as:

$$g_b(x_1) = \{0, 0.1, 0.5, 0.2\}, g_b(x_2) = \{0.1, 0, 0.6, 0.3\}, g_b(x_3) = \{0.5, 0.6, 0, 0.3\}, g_b(x_4) = \{0.2, 0.3, 0.3, 0\}. \tag{8}$$

The sample set $X = \{x_1, x_2, x_3, x_4\}$ is granulated on feature c as:

$$g_c(x_1) = \{0, 0.1, 0.3, 0.5\}, g_c(x_2) = \{0.1, 0, 0.4, 0.4\}, g_c(x_3) = \{0.3, 0.4, 0, 0.8\}, g_c(x_4) = \{0.5, 0.4, 0.8, 0\}. \tag{9}$$

The feature granular vectors of the sample $X = \{x_1, x_2, x_3, x_4\}$ are:

$$G_{\{a,b,c\}}(x_1) = (g_a(x_1), g_b(x_1), g_c(x_1))^T = (\{0, 0.65, 0.6, 0.3\}, \{0, 0.1, 0.5, 0.2\}, \{0, 0.1, 0.3, 0.5\})^T, \tag{10}$$

$$G_{\{a,b,c\}}(x_2) = (g_a(x_2), g_b(x_2), g_c(x_2))^T = (\{0.65, 0, 0.05, 0.35\}, \{0.1, 0, 0.6, 0.3\}, \{0.1, 0, 0.4, 0.4\})^T, \tag{11}$$

$$G_{\{a,b,c\}}(x_3) = (g_a(x_3), g_b(x_3), g_c(x_3))^T = (\{0.5, 0.6, 0, 0.3\}, \{0.5, 0.6, 0, 0.3\}, \{0.3, 0.4, 0, 0.8\})^T, \tag{12}$$

$$G_{\{a,b,c\}}(x_4) = (g_a(x_4), g_b(x_4), g_c(x_4))^T = (\{0.3, 0.35, 0.3, 0\}, \{0.2, 0.3, 0.3, 0\}, \{0.5, 0.4, 0.8, 0\})^T. \tag{13}$$

The decision set $Y = \{x_1, x_2, x_3, x_4\}$ is extended and granularized as

$$Y\{x_1\} = \{22.5, 22.5, 22.5, 22.5\}, \tag{14}$$

$$Y\{x_2\} = \{33, 33, 33, 33\}, \tag{15}$$

$$Y\{x_3\} = \{45, 45, 45, 45\}, \tag{16}$$

$$Y\{x_4\} = \{10.05, 10.05, 10.05, 10.05\}. \tag{17}$$

Table 1. A data set.

X	a	b	c	D
x_1	1	0.5	0.6	22.5
x_2	0.35	0.4	0.5	33
x_3	0.4	1	0.9	45
x_4	0.7	0.7	0.1	10.05

3. Granular Elastic Network Regression

3.1. Granular Operations and Measures

To construct the granular regression system, in this subsection, we give the granular operators and measures.

Definition 5. *Let $g_a(x)$, $g_b(x)$ be the two granules of sample x on features a, b, respectively. The addition, subtraction, multiplication, and division operations for the granules are:*

$$g_a(x) + g_b(x) = \{g_a(x)_j + g_b(x)_j\}_{j=1}^k, \tag{18}$$

$$g_a(x) - g_b(x) = \{g_a(x)_j - g_b(x)_j\}_{j=1}^k, \tag{19}$$

$$g_a(x) * g_b(x) = \{g_a(x)_j * g_b(x)_j\}_{j=1}^k, \tag{20}$$

$$g_a(x) / g_b(x) = \{g_a(x)_j / g_b(x)_j\}_{j=1}^k. \tag{21}$$

Definition 6. *Let $g_a(x)$, $g_a(y)$ be the two granules of samples x and y on features a. The addition, subtraction, multiplication, and division operations for the granules are:*

$$g_a(x) + g_a(y) = \{g_a(x)_j + g_a(y)_j\}_{j=1}^k, \tag{22}$$

$$g_a(x) - g_a(y) = \{g_a(x)_j - g_a(y)_j\}_{j=1}^k, \tag{23}$$

$$g_a(x) * g_a(y) = \{g_a(x)_j * g_a(y)_j\}_{j=1}^k, \tag{24}$$

$$g_a(x) / g_a(y) = \{g_a(x)_j / g_a(y)_j\}_{j=1}^k. \tag{25}$$

The result of four operations with two granules is also a granule. The operations in Definition 5 are granular operations for granules of the same sample on different features, and the operations in Definition 6 are granular operations for granules of different samples on the same feature.

Definition 7. *Let the granular vector be $G(x_j) = (g_1(x)_j, g_2(x)_j, g_3(x)_j, \ldots, g_m(x)_j)^T$, $G(x_i) = (g_1(x)_i, g_2(x)_i, g_3(x)_i, \ldots, g_m(x)_i)^T$, the dot product of these two granular vectors is defined as:*

$$G(x_i) \cdot G(x_j) = g_1(x_i) * g_1(x_j) + g_2(x_i) * g_2(x_j) + g_3(x_i) * g_3(x_j) + \ldots + g_m(x_i) * g_m(x_j). \tag{26}$$

The dot product of two granular vectors outcomes to a granule.

Definition 8. *Let the granule be $g(x) = \{r_j\}_{j=1}^k$, whose size is:*

$$size(g(x)) = |g(x)| = \sum_{j=1}^k r_j \tag{27}$$

Definition 9. *Let the granule be $g(x) = \{r_j\}_{j=1}^k$, and its granular norms as follows:*

(a). The 1-norm of the granule:

$$Norm-1(g(x)) = ||g(x)||_1 = \sum_{j=1}^{k} |r_j|, \tag{28}$$

(b). The 2-norm of the granule:

$$Norm-2(g(x)) = ||g(x)||_2 = \sqrt{\sum_{j=1}^{k} r_j{}^2}, \tag{29}$$

(c). The p-norm of the granule:

$$Norm-p(g(x)) = ||g(x)||_p = \left(\sum_{j=1}^{k} r_j{}^p\right)^{\frac{1}{p}}. \tag{30}$$

Definition 10. *Let the granular vector be $g(x) = \{r_j\}_{j=1}^{k}$, and its granular vector norms as follows:*
(a). The 11-norm of the granular vector:

$$Norm-11(G(x_i)) = ||G(x_i)||_{11} = |\sum_{j=1}^{i} \sum_{j=1}^{k} |r_j||, \tag{31}$$

(b). The 12-norm of the granular vector:

$$Norm-12(G(x_i)) = ||G(x_i)||_{12} = \sum_{j=1}^{i} \sqrt{\sum_{j=1}^{k} r_j{}^2}, \tag{32}$$

(c). The 21-norm of the granular vector:

$$Norm-21(G(x_i)) = ||G(x_i)||_{21} = \sqrt{\sum_{j=1}^{i} \left(\sum_{j=1}^{k} r_j\right)^2}, \tag{33}$$

(d). The 22-norm of the granular vector:

$$Norm-22(G(x_i)) = ||G(x_i)||_{22} = \sqrt{\sum_{j=1}^{i} \sqrt{\sum_{j=1}^{k} r_j{}^2}}. \tag{34}$$

3.2. Granular Elasticity Regression Model

In the previous subsection, we defined several granular operations and granular norms. In the following, we construct our granular elastic network regression model with the defined granular operations.

Definition 11. *In the regression data set $U = (X, C, D)$, $x \in X$ is granularized and expanded according to Definition 1 as $G(x) = (g_1(x), g_2(x), g_3(x), \ldots, g_m(x), 1)$, and the weight granular vector is $W = (w_1, w_2, w_3, \ldots, w_m, b)$. The granular regression equation is defined as:*

$$Reg(x) = W \cdot G(x) = w_1 * g_1(x) + w_2 * g_2(x) + \ldots + w_m * g_m(x) + 1 * b. \tag{35}$$

Similar to the traditional linear regression weight randomization, the granular regression network also uses the normalized distribution to randomly initialize the granular weights.

In the traditional regression model, lasso regression can make the weights sparse, while ridge regression can shrink the weight coefficients to make the weights average. By combining

the L1 penalty with the L2 penalty, the elastic network is able to both sparse the weights and shrink the weight coefficients. Similarly, in the granular regression system the granular elastic network uses a granular vector 11-norm penalty and a granular vector 22-norm penalty. The loss function of granular elastic network regression is defined as follows.

Definition 12. *In the regression data set U = (X,C,D), the weight granular vector is $W = (w_1, w_2, \ldots, w_m, b)$, the granular vector is $G(x) = (g_1(x), g_2(x), g_3(x), \ldots, g_m(x), 1)$ and the decision granular vector is $G(y) = (g_1(y), g_2(y), g_3(y), \ldots, g_m(y), 0)$. The granular loss function is defined as*

$$G(e) = \frac{1}{2}||W \cdot G(x) - \frac{1}{m}G(y)||_2^2 + \alpha||W||_{11} + \frac{1}{2}\beta||W||_{22}^2, \tag{36}$$

where $W \cdot G(x) - \frac{1}{m}G(y)$ is a granule.

We have the derivative operation for $G(e)$:

$$\frac{\partial G(e)}{\partial w} = -1 * G(x) * (\frac{1}{m}G(y) - W \cdot G(x)) + \alpha * sign(W) + \beta * W. \tag{37}$$

3.3. Granular Elastic Network Optimization Algorithm

In order to optimize the granular regression elastic network to attain better regression results, we propose the granular elastic regression optimizer in the following:

$$W_c = \underset{W_c}{\arg\min}G(e) = \underset{W_c}{\arg\min}\frac{1}{2}||W \cdot G(x) - \frac{1}{m}G(y)||_2^2 + \alpha||W||_{11} + \frac{1}{2}\beta||W||_{22}^2, \tag{38}$$

We use stochastic gradient descent to solution the granular elasticity network parameters. To obtain the granular regression error, we define the root mean square error *RMSE* of the granules expressed as follows:

$$GRMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left|\left|W_c \bullet G(x_i) - \frac{1}{m}G(y_i)\right|\right|_2^2} \tag{39}$$

The details of the stochastic gradient descent optimized granular elastic network regression algorithm are shown in Algorithm 1. In the training algorithm, *m* represents the maximum number of training iterations. *η* is the learning rate.

Algorithm 1 Granular elastic network optimization algorithm

Input: The training set is $U = (X, Y)$, where is the feature vector, $x_i \in X$, is the regression decision vector $y_i \in Y, i = 1, 2, ..., n$; the learning rate is *η*, and the number of iterations *m*
Output: Granular weight matrix *W*.
(1) The sample set *X* is granularized over the feature set (X, C) to obtain $G(X)$
(2) The decision set Y is extended granularly as $G(Y)$
(3) Construct a granular elastic network and randomly initialize the granular weights in the network
(4) $i = 0$
(5) for *i* to *m*
(6) $GRMSE^i = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left|\left|W_c \bullet G(x_i) - \frac{1}{m}G(y_i)\right|\right|_2^2}$
(7) $W_c^{i+1} = W_c^i - \eta\frac{\partial G(e)}{\partial w}$
(8) $GRMSE^{i+1} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left|\left|W_c^{i+1} \bullet G(x_i) - \frac{1}{m}G(y_i)\right|\right|_2^2}$
(9) $\Delta RMSE = GRMSE^i - GRMSE^{i+1}$
(10)   if $\Delta RMSE > \xi$
(11)   break

---

4. Experimental Analysis

Multiple UCI datasets including Estates, Concrete, QSAR fish, Airfoil, Energy, Yacht, etc., are used for comparison experiments. Further, in order to comprehensively analyze the proposed method, we select the Estates, Concrete, and QSAR fish datasets for hyperparameter analysis and fitting analysis. The Concrete dataset consists of 8 conditional attributes and 1 decision attribute, the Estates dataset consists of 9 conditional attributes and 1 decision attribute, and the QSAR fish dataset consists of 6 conditional attributes and 1 decision attribute. We randomly select 80% of the samples from the dataset as the training set and the remaining 20% as the test set. The random number seed is fixed to ensure that the results of each selection are the same.

We measure the model by mean absolute error (MAE) and the coefficient of determination (R^2). MAE is expressed as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} ||y_i' - y_i||, \tag{40}$$

where y_i' denotes the predicted value and y_i denotes the true value. The smaller the MAE, indicates the smaller the average prediction error of the model. In order to represent R^2, we need to represent the concepts of sum of squares for regression SSR, sum of squares for error SSE, and sum of squares for total SST. SSR is expressed as follows:

$$SSR = \sum_{i=1}^{n} (y_i' - \overline{y_i})^2, \quad where \quad \overline{y_i} = \frac{\sum_{i=1}^{n} y_i}{n}. \tag{41}$$

SSE is represented as:

$$SSE = \sum_{i=1}^{n} (y_i' - y_i)^2. \tag{42}$$

SST is defined as:

$$SST = SSR + SSE = \sum_{i=1}^{n} (y_i - \overline{y_i})^2. \tag{43}$$

R^2 is calculated by

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n} (y_i' - y_i)^2}{\sum_{i=1}^{n} (y_i - \overline{y_i})^2}. \tag{44}$$

For the test set, the larger R^2 model indicates the better fit, the greater interpretation of the dependent variable by the variables and the high percentage of the total variation caused by the independent variables. The maximum value of R^2 is 1.

4.1. Convergence Analysis

We select two kinds of learning rate for convergence analysis. First of all, we set the learning rate of the algorithm to 0.001 and the loss function to MSE. Figures 1–3 show the training processes of the granular elastic network on the Concrete, Estate, and QSAR datasets when learning the rate is 0.001, respectively. The granular elastic network converges on all three datasets. When the number of iteration rounds is 800, the granular elastic network reaches convergence on the Estate and QSAR datasets with smooth loss curves. For the Concrete data set, the granular elastic network also converges, but the loss curve is more volatile.

Then, the learning rate is adjusted to 0.005. Figures 4–6 show the training processes of granular elastic network when the learning rate is 0.005. For Concrete dataset, it can be seen from Figures 1 and 4 that the convergence curve when the learning rate is 0.005 is smoother than that when the learning rate is 0.001. However, for QSAR data set and Estate data set, the training curves are tortuous when the learning rate is 0.005, but they are able to convergence.
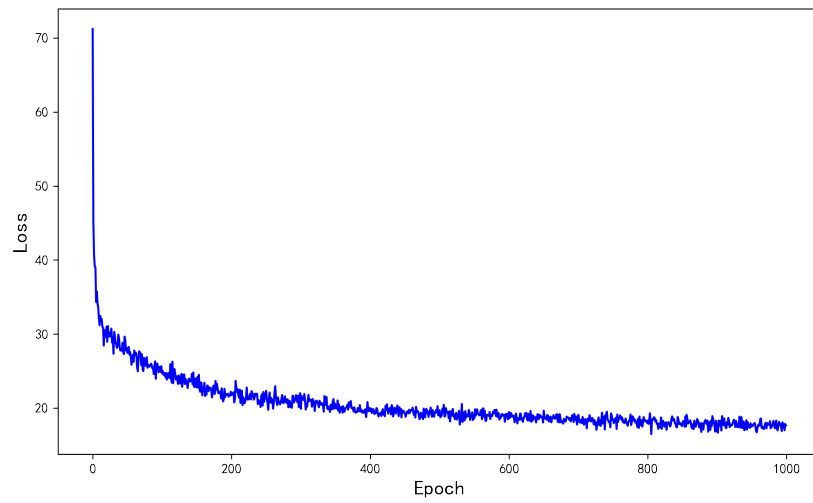
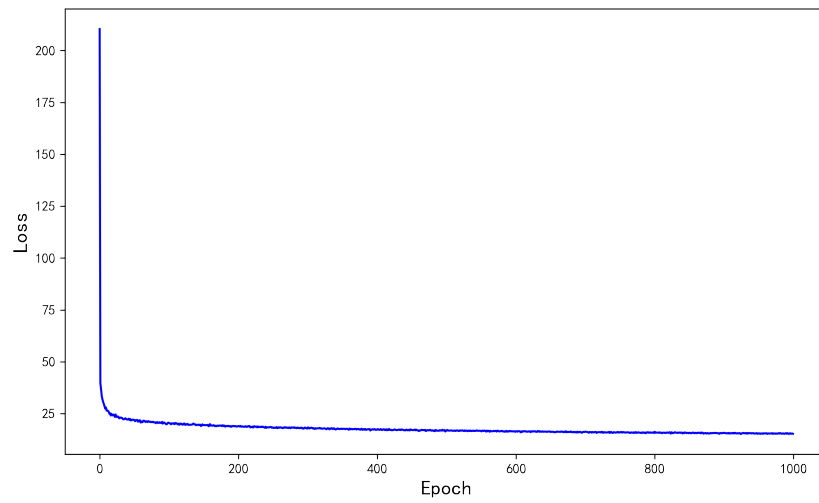**Figure 1.** Training process for Concrete dataset when learning the rate is 0.001.
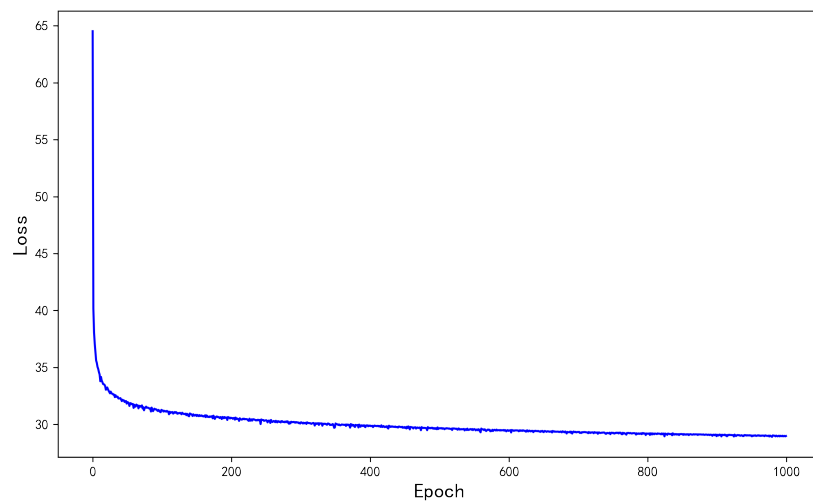
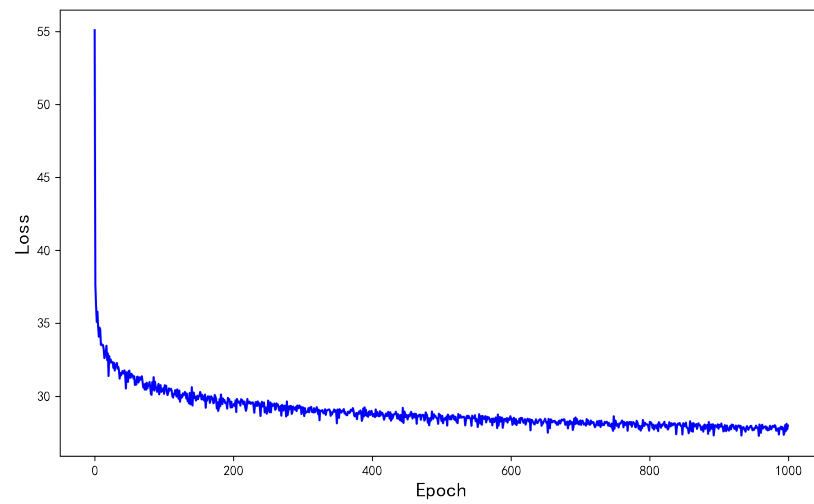**Figure 2.** Training process for Estate dataset when learning the rate is 0.001.
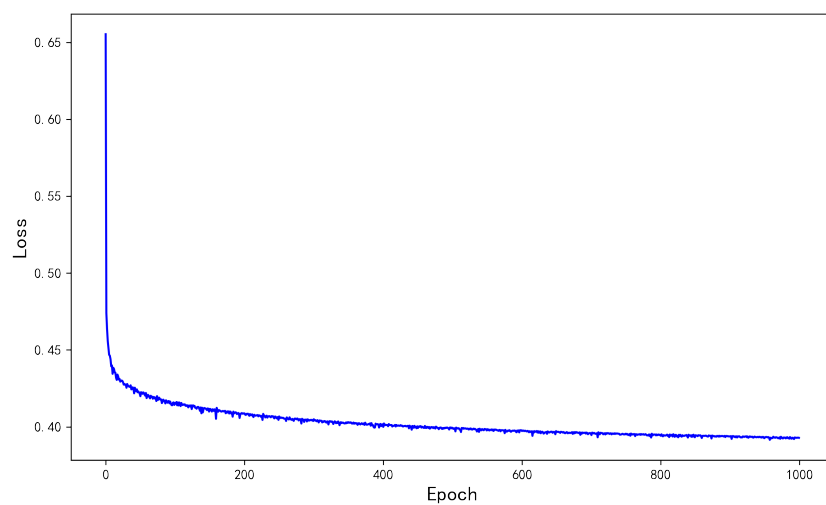
Figure 3. Training process for QSAR dataset when learning the rate is 0.001.
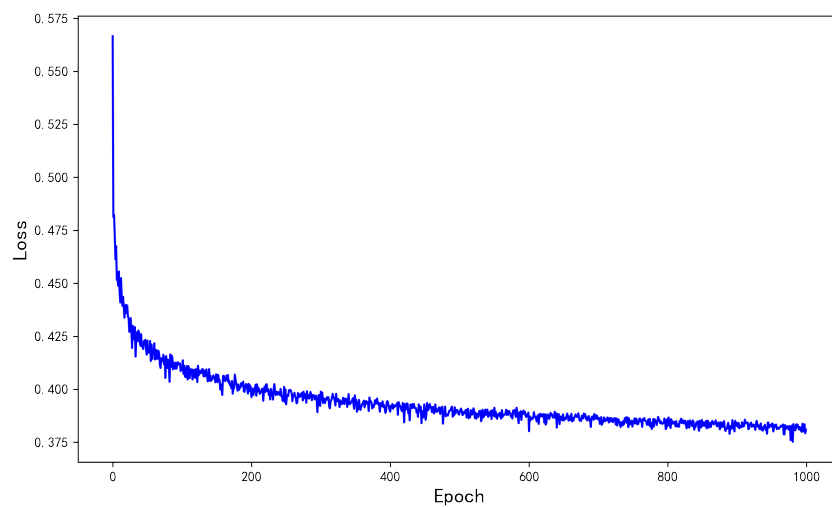
Figure 4. Training process for Concrete dataset when learning the rate is 0.005.

Figure 5. Training process for Estate dataset when learning the rate is 0.005.

Figure 6. Training process for QSAR dataset when learning the rate is 0.005.

4.2. α and β Penalty Coefficient Proportional Impacts

In this subsection we show the effect of the α and β ratios on the different test sets. To facilitate the analysis, we set $\alpha + \beta = 1$, and the L11 ratio= $\frac{\alpha}{\alpha+\beta}$. L11 ratio indicates the ratio of L11 penalty to L11 and L22 combined penalty. Smaller L11 ratio means smaller L11 penalty to L11 and L22 combined penalty. Larger L11 ratio means smaller L22 penalty to L11 and L22 combined penalty.

For the Concrete test set, the effect of L11 ratio on the model is shown in Figure 7. It can be seen from Figure 7 that the MAE is minimum when the L11 ratio = 0.5. When L11 ratio = 0.8, the MAE is maximum and reaches about 4.894. From Figure 8, it can be seen that when L11 ratio = 0.9, the MAE of the estate test set is minimum. When L11 ratio = 0.6, the MAE of the estate test set is the smallest. For the QSAR test set, as shown in Figure 9, the MAE reaches a minimum of about 0.799 at L11 ratio = 0.8, while the MAE of the QSAR test set reaches a maximum of about 0.831 at L11 ratio = 0.2.
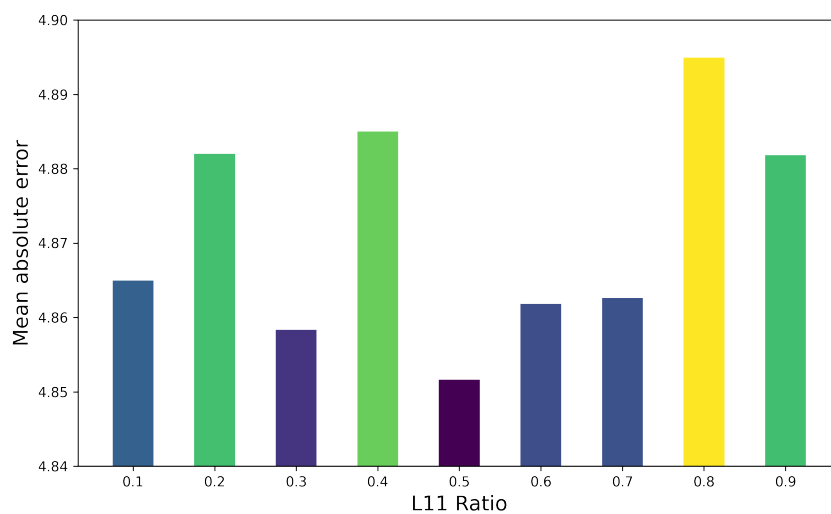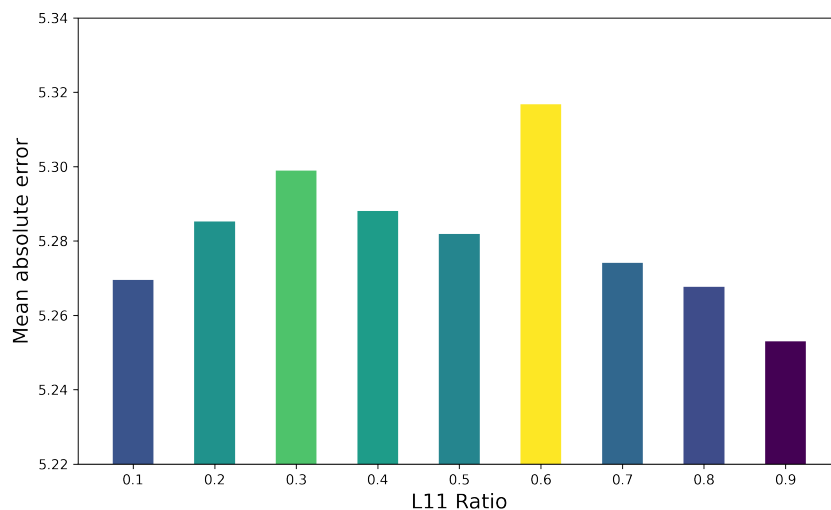

Figure 7. Impacts of L11 ratio on Concrete dataset.

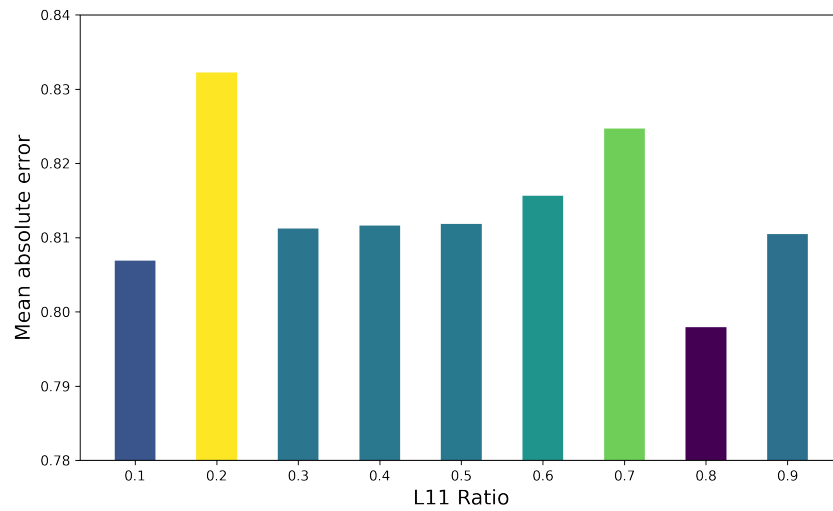**Figure 8.** Impacts of L11 ratio on Estate dataset.

Figure 9. Impacts of L11 ratio on QSAR dataset.

4.3. Fitting Analysis

To better show the regression results, we selected 50 samples from the test set to plot the regression fitting curves, as shown in Figures 10–12. Figure 10 represents the fitting curves of the granular elastic network in the Concrete test set. From Figure 10, we can see that the granular elastic network has good regression accuracy in the first 15 samples, but the regression error is larger in the last 10 samples. For the Estate test set, the first 5 samples are fitted very accurately, but samples 9 to 14 are fitted poorly. Sample 35 to final sample 50 are also fitted with less error. Comparing Figures 10 and 12, it can be seen that the granular elastic network does not fit as well on the QSAR dataset as it does on the Concrete dataset.
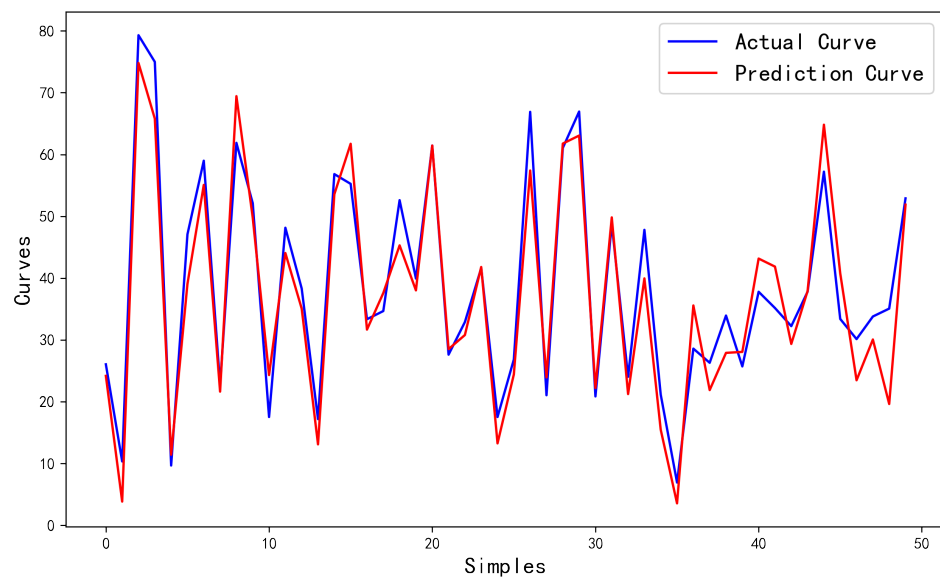

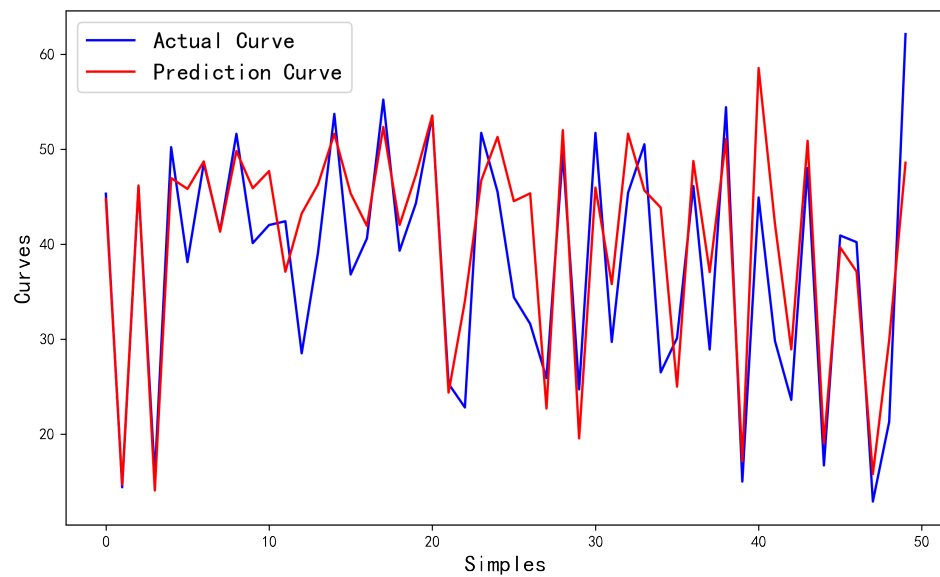**Figure 10.** The regression fitting curve for Concrete test set.

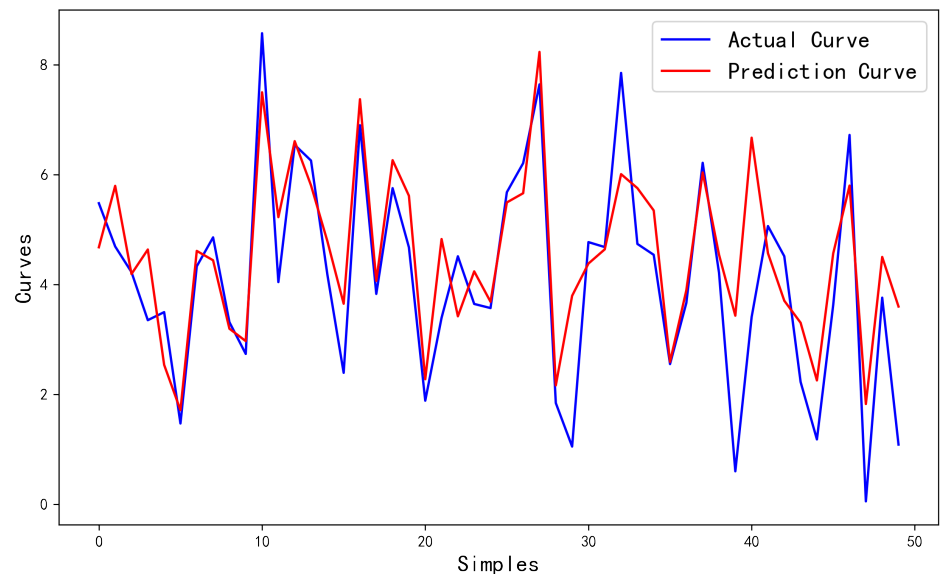**Figure 11.** The regression fitting curve for Estate test set.

Figure 12. The regression fitting curve for QSAR test set.

4.4. Comparison of Granular Elastic Network Regression and Classical Regression Algorithms

In order to illustrate the effectiveness of the proposed method, the comparative experiments of granular elastic network and several classical regression algorithms in several UCI data sets are shown below. We use mean absolute error (MAE) and the coefficient of determination (R^2) to measure model fitting performance. For ridge regression, lasso regression, elastic network regression, and granular elastic network regression all 5-fold cross-validation are performed on the training set to extract regular coefficients with the best results, and then substituted into the test set for testing. We bold the best results in Tables 2 and 3.

Table 2 shows the MAE results on the test sets of granular elastic network and traditional regression networks. Smaller MAE means smaller regression error. Except for the Daily demand dataset and the QSAR dataset, the granular elastic network has a smaller mean absolute error (MAE) than the classical regression algorithm. Especially in the Con-

crete, Energy and Yacht datasets, the MAE of the granular elastic network is particularly small compared to the classical regression algorithms.

Table 2. The *MAE* results for granular elastic network regression and classical regression algorithms.

Datasets	Linear Regression	Ridge	Lasso	Elastic Network	Granular Elastic Network
Concrete	8.1142	8.0997	8.0997	8.0978	**4.3481**
Estate	6.2929	6.2205	6.2284	6.2346	**5.1689**
QSAR	0.777	**0.769**	0.7765	0.7727	0.7911
Airfoil	3.5452	3.5472	3.5477	3.5621	**3.2883**
Ale	0.223	0.2217	0.2179	0.2169	**0.1471**
Slump	2.7413	2.7503	2.7543	2.7615	**1.9253**
Daliy demand	**71.8545**	80.9481	71.8734	77.0038	85.7092
Boston	3.9358	3.926	3.917	3.8874	**3.0508**
Energy	2.2216	2.2439	2.2582	2.3377	**0.9997**
Yacht	8.3883	7.581	8.3399	7.462	**1.9873**

Table 3. The coefficient of determination results for granular elastic network regression and classical regression algorithms.

Datasets	Linear Regression	Ridge	Lasso	Elastic Network	Granular Elastic Network
Concrete	0.3937	0.3904	0.3929	0.3906	**0.8842**
Estate	0.5289	0.4995	0.5128	0.5059	**0.6303**
QSAR	0.3637	0.331	0.3626	0.3499	**0.4113**
Airfoil	0.1676	0.1621	0.1598	0.1257	**0.3566**
Ale	0.1009	0.0937	0.1011	0.0829	**0.7212**
Slump	0.791	0.7865	0.7926	0.7816	**0.9125**
Daliy demand	**0.5698**	0.4637	0.5692	0.4466	0.3353
Boston	0.3531	0.3455	0.3437	0.3029	**0.5521**
Energy	0.893	0.8917	0.8906	0.8749	**0.9844**
Yacht	0.2779	−0.1795	0.2705	-1.3873	**0.9299**

The coefficient of determination (R^2) of the granular elastic network and the traditional regression algorithms on the test sets are shown in Table 3. A higher coefficient of determination (R^2) means better regression fitting performance. The maximum value of R^2 is 1. In the vast majority of datasets, the R^2 of the proposed algorithm is better than the classical regression algorithms. In the Yacht dataset, even though the R^2 of some traditional algorithms is negative, the R^2 of the granular elastic network is as high as about 0.93.

In general, the granular elastic network regression algorithm proposed in this paper has better regression performance than the classical regression algorithm.

5. Conclusions and Discussion

Traditional regression algorithms can only be mostly applied only to real-valued data, but not to fuzzy and aggregated domains. We design relevant granulation methods and granulation algorithms for fuzzy and ensemble data. Then, a novel granular regression system is proposed in this paper and two regularization penalties are added to it. Similarly, the proposed granular elastic network optimization algorithm is applied to granular elastic network learning. Finally, we conducted experiments on the UCI datasets and compared it with several traditional regression algorithms. The experimental results show that the granular elastic network proposed in this paper has the advantage of smaller regression error and better fit compared to traditional regression algorithms.

As the granulation method in this paper is global granulation, it is difficult to process big data in this granular regression method. We will propose a new granulation approach to apply granular regression to big data. In future work, we will apply the granular regression

system to the nonlinear regression system. We will also propose new granulation methods to further improve the regression performance.

References

1. Zadeh, L.A. Fuzzy sets and information granularity. *Adv. Fuzzy Set Theory Appl.* **1979**, *11*, 3–18.
2. Hobbs, J.R. Granularity. In Proceedings of the IJCAI, Los Angeles, CA, USA, 18–23 August 1985; pp. 432–435.
3. Zadeh, L.A. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets Syst.* **1997**, *90*, 111–127. [CrossRef]
4. Lin, T.Y. Data Mining: Granular Computing Approach. *Lect. Notes Comput. Sci.* **1999**, *1574*, 24–33.
5. Lin, T.Y.; Zadeh, L.A. Special issue on granular computing and data mining. *Int. J. Intell. Syst.* **2004**, *19*, 565–566. [CrossRef]
6. Bazan, J.G.; Nguyen, H.S.; Nguyen, S.H.; Synak, P.; Wróblewski, J. Rough Set Algorithms in Classification Problem. In *Rough Set Methods and Applications*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 49–88.
7. Liu, Q.; Jiang, S.L. Reasoning about information granules based on rough logic. *Lect. Notes Comput. Sci.* **2002**, *2475*, 139–143.
8. Yao, Y. Information granulation and rough set approximation. *Int. J. Intell. Syst.* **2001**, *16*, 87–104. [CrossRef]
9. Miao, D.Q.; Fan, S. The calculation of knowledge granulation and its application. *Syst. Eng.-Theory Pract.* **2002**, *22*, 48–56.
10. Miao, D.Q.; Xu, F.F.; Yao, Y.Y.; Wei, L. Set-theoretic formulation of granular computing. *Chin. J. Comput.* **2012**, *35*, 351–363. [CrossRef]
11. Wang, G.Y.; Zhang, Q.H.; Ma, X.A.; Yang, Q.S. Granular computing models for knowledge uncertainty. *J. Softw.* **2011**, *22*, 676–694. [CrossRef]
12. Chen, Y.; Miao, D.Q.; Wang, R.; Wu, K. A rough set approach to feature selection based on power set tree. *Knowl.-Based Syst.* **2011**, *24*, 275–281. [CrossRef]
13. Qian, J.; Miao, D.; Zhang, Z.; Yue, X. Parallel attribute reduction algorithms using MapReduce. *Inform. Sci.* **2014**, *279*, 671–690. [CrossRef]
14. Chen, Y.M.; Qin, N.; Li, W.; Xu, F.F. Granule structures, distances and measures in neighborhood systems. *Knowl.-Based Syst.* **2019**, *165*, 268–281. [CrossRef]
15. Li, W.; Chen, Y.M.; Song, Y.P. Boosted K-nearest neighbor classifiers based on fuzzy granules. *Knowl.-Based Syst.* **2020**, *195*, 105–606. [CrossRef]
16. Chiaselotti, G.; Gentile, T.; Infusino, F. Granular computing on information tables: Families of subsets and operators. *Inform. Sci.* **2018**, *442*, 72–102. [CrossRef]
17. Lei, T.; Jia, X.; Zhang, Y.; Liu, S.; Meng, H.; Nandi, A.K. Superpixel-Based Fast Fuzzy C-Means Clustering for Color Image Segmentation. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 1753–1766. [CrossRef]
18. Fujita, H.; Gaeta, A.; Loia, V.; Orciuoli, F. Resilience analysis of critical infrastructures: A cognitive approach based on granular computing. *IEEE Trans. Cybern.* **2019**, *49*, 1835–1848. [CrossRef]
19. Chen, Y.M.; Miao, D.Q.; Wang, R.Z. A rough set approach to feature selection based on ant colony optimization. *Pattern Recognit. Lett.* **2010**, *31*, 226–233. [CrossRef]
20. Hu, Q.H.; Yu, D.R.; Liu, J.F.; Wu, C.X. Neighborhood rough set based heterogeneous feature subset selection. *Inf. Sci.* **2008**, *178*, 3577–3594. [CrossRef]
21. Li, D.F. TOPSIS-based nonlinear-programming methodology for multiattribute decision making with interval-valued intuitionistic fuzzy sets. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 391. [CrossRef]
22. Lin, G.P.; Liang, J.Y.; Li, J.J. A fuzzy multigranulation decision-theoretic approach to multi-source fuzzy information systems. *Knowl.-Based Syst.* **2016**, *91*, 102–113. [CrossRef]
23. Mendel, J.M.; Bonissone, P.P. Critical Thinking about Explainable AI (XAI) for Rule-Based Fuzzy Systems. *IEEE Trans. Fuzzy Syst.* **2021**, *29*, 3579–3593. [CrossRef]

24. Cosme, L.B.; Caminhas, W.M.; Dangelo, M.F.; Palhares, R.M. A novel fault-prognostic approach based on interacting multiple model filters and fuzzy systems. *IEEE Trans. Ind. Electron.* **2018**, *66*, 519–528. [CrossRef]

25. Wang, L.; Dong, J. Adaptive Fuzzy Consensus Tracking Control for Uncertain Fractional-Order Multiagent Systems With Event-Triggered Input. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 310–320. [CrossRef]

26. Hu, H.; Pang, L.; Tian, D.; Shi, Z. Perception granular computing in visual haze-free task. *Expert Syst. Appl.* **2014**, *41*, 2729–2741. [CrossRef]

27. Naseem, I.; Togneri, R.; Bennamoun, M. Linear regression for face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 2106–2112. [CrossRef]

28. Chen, Y.R.; Rezapour, A.; Tzeng, W.G. Privacy-preserving ridge regression on distributed data. *Inf. Sci.* **2018**, *451*, 34–49. [CrossRef]

29. Tibshirani, R.J. Regression shrinkage and selection via the LASSO. *J. R. Statist. Soc.* **1996**, *73*, 273–282. [CrossRef]

30. Zhang, Z.; Lai, Z.; Xu, Y.; Shao, L.; Wu, J.; Xie, G.S. Discriminative elastic-net regularized linear regression. *IEEE Trans. Image Process.* **2017**, *26*, 1466–1481. [CrossRef]